Progress In Electromagnetics Research M, Vol. 11, 225–240, 2010

OPTIMIZED SIMULATION ALGORITHMS FOR FRACTAL GENERATION AND ANALYSIS

B. Camps-Raga and N. E. Islam

University of Missouri
Columbia, MO 65211, USA

Abstract—A set of algorithms, specifically developed to facilitate an effective modeling of fractal-boundary microstrip antennas in the analysis of such structures through numerical electromagnetic (EM) solvers is presented in this paper. A fractal generator based on the implementation of an Iterated Function System (IFS) produces the geometry specified in accordance with the user-defined input parameters. The structure is created through a solver-specific interface and is thus applicable to a commercially available EM simulation suite. The generation of specific shapes through these algorithms provides a flexible method to study different geometries without the need to modify either the interface or the solver. Three structures based on the Minkowski fractal obtained through these techniques have been studied using two EM solvers for comparison. The frequency-domain results show good agreement between the two solvers, thus validating the algorithms implemented. Complex structures with higher iterations can be studied using these algorithms.

1. INTRODUCTION

Methods for refining simulation algorithms to analyze electromagnetic problems involving complex structures have been of interest to engineers and scientists for some time. One such algorithm that requires minor modifications is the generation of structures for antenna simulation using the finite-difference time-domain (FDTD) method [1]. For example, in case of microstrip antennas with patches having fractal-like boundaries, analytical methods are not only difficult to apply but also inaccurate when the geometry is not well defined. For higher iterations of fractal geometries, the structures become increasingly complex and extremely difficult to draw without a known

Corresponding author: B. Camps-Raga (bcampsr@gmail.com).

mathematical algorithm to define each iteration of the fractal. A faithful reproduction of the geometry is also a necessity as the structure is scaled down. In this paper we first introduce an algorithm to specifically define the geometry for a Minkowski fractal, and then we use the output to model fractal microstrip antennas. The fractal geometry has been interfaced with both an in-house FDTD code and the commercial electromagnetic simulation suite CST Microwave Studio®. Since the use of higher iterations has been shown to allow antenna miniaturization [2], the performance of these algorithms is tested by comparing the results obtained for a square patch and the first two Minkowski iterations derived from it.

Following this brief introduction, we have divided the rest of the paper as follows: a general overview of this fractal series is given in Section 2, where the algorithm referred to as 'Minkowski fractal algorithm (MFA)' is discussed. A description of the interfaces developed to implement the geometry under analysis (including the conducting patch obtained through the MFA) is given in Section 3. Section 4 addresses several design considerations in the FDTD code, and the simulation results for three types of microstrip patches are discussed in Sections 5 and 6. Finally, we conclude in Section 7.

2. MINKOWSI FRACTAL ALGORITHM (MFA)

Fractals, as defined by Mandelbrot [3], are recursively generated geometries with self-similarities and fractional dimensions. There are different mechanisms to generate fractals. The algorithms developed in this work are based on a subset of fractals, referred to as deterministic, which can be obtained using scaled-down and rotated copies of an initial geometry. Thus a recursive algorithm based on a deterministic fractal can be developed to generate any occurrence of such fractal for a given number of iterations. For example, the first four iterations of the Minkowski fractal, shown in Fig. 1, can be derived from the square patch through a specific algorithm.

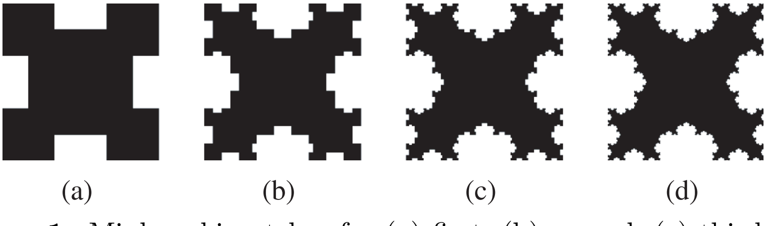


Figure 1. Minkowski patches for (a) first, (b) second, (c) third, and (d) fourth iterations.

The MFA can be implemented through the three parameters shown in Fig. 2(a). Here, a is the length of the generator, b is the indentation width, and h is the indentation depth. For a first iteration, a is set equal to the length of the initiator (Fig. 2(b)), b is chosen as a fraction of a, and h is related to a through the indentation factor, ρ:

$$\rho = \frac{h}{a/3} \tag{1}$$

The choice of parameters b and ρ determines the overall area that is removed off the patch, and thus its fractal dimension [4], which is a measure of its space-filling properties and complexity mathematically related to multi-resonant operation [5]. This number can be calculated,

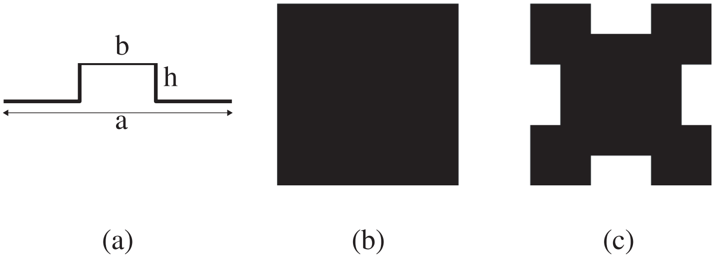


Figure 2. Generation of a Minkowski fractal patch: (a) generator, (b) initiator, (c) first iteration (M_1).

Table 1. Fractal dimension for different values of the indentation factor.

ρ	Fractal dimension
0.10	1.05
0.20	1.10
0.30	1.14
0.40	1.18
0.50	1.22
0.60	1.27
0.70	1.32
0.80	1.36
0.90	1.41

for a general case, by solving the following equation [6]:

$$k_1 \left(\frac{1}{h_1} \right)^D + k_2 \left(\frac{1}{h_2} \right)^D + \ldots + k_n \left(\frac{1}{h_n} \right)^D, \tag{2}$$

where k_n is the number of copies of the initiator scaled by h_n. As also discussed in [6], the fractal dimension of a fractal patch must be calculated based on the geometry of one side only, since the self-similarity is valid for straight segments and not for closed structures. Using this approach, the fractal dimension of a given iteration of a Minkowski patch can be found by considering the initiator as a segment of length a, equal to the side of the quad. For example, the fractal dimension, D, for a first Minkowski iteration, is obtained from

$$k_1 \left(\frac{1}{h_1} \right)^D + k_2 \left(\frac{1}{h_2} \right)^D = 1, \tag{3}$$

where $k_1 = 3$ and $h_1 = 3$ represent the three horizontal segments, while $k_2 = 2$ and $h_2 = 3/\rho$ represent the vertical segments (five segments in total), and the indentation length, b, has been fixed to one third of a. The solution of (3), for several values of the indentation factor, yields the fractal dimensions shown in Table 1. The MFA has been implemented from an iterated function system (IFS) specifically defined for this geometry. An IFS is defined as the recursive application of N affine transformations $\{w_n,\ n = 1, \ldots, N\}$ to an initial geometry [7, 8]. Following the discussion presented in [9], this can be expressed, using matrix notation, as:

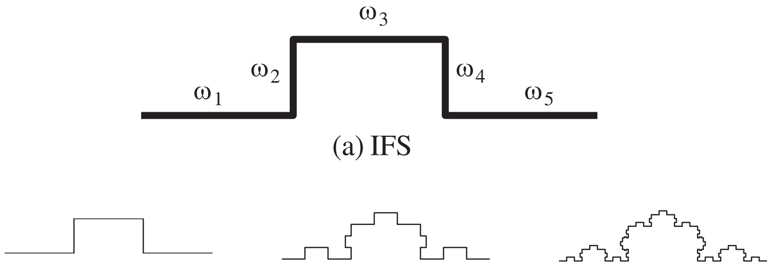
$$w_n(x, y) = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix}, \tag{4}$$

where a, b, c, and d control rotation and scaling, while e and f control linear translation, and all six parameters are real numbers. Considering an initial geometry A (a segment of a given length, for example) and a set of linear transformations $\{w_i,\ i = 1, \ldots, N\}$, a first iteration is produced by first applying each transformation w_i to the initial geometry A, and then combining the resulting geometries $w_i(A)$ by

$$W(A) = \bigcup_{i=1}^{N} w_i(A), \tag{5}$$

where W is the Hutchinson operator [8]. The fractal geometry is then obtained by repeatedly applying this operator to the output of the previous step. This is illustrated in [9] for a Koch curve. In this work, an IFS has been defined and implemented to obtain the Minkowski fractal which is used to form the closed patch. The iterative process involved in the generation of these fractals is described briefly as follows. For the generation of the desired fractal geometry based on the Minkowski island, the IFS technique is applied by considering the initiator shown in Fig. 2(a), with parameters a, b, and h (as a function of ρ). A segment of length a is chosen as the original geometry, and five transformations $\{w_1, \ldots, w_5\}$ are applied to obtain the five segments labeled in Fig. 3(a). The same process can be recursively applied to each resulting segment to yield the next iteration. As an example, the first three iterations obtained following this method are also depicted in Fig. 3 for $a = 1$, $b = 0.4$, and $\rho = 0.5$. The corresponding Minkowski patch is achieved by applying rotation, translation and symmetry to form a closed geometry.

The algorithm produces a list of points containing the coordinates of all vertices in the geometry. Special care is taken to avoid redundant points which result from applying the aforementioned rotation, translation and symmetry transformations. This list is then stored for further processing as needed. The four fractal patches shown in Fig. 4 show the effect of ρ in the final geometry. In order to compare



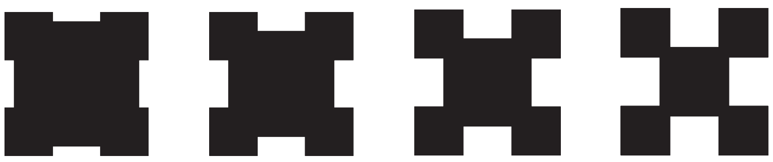
(a) IFS

(b) Iteration 1

(c) Iteration 2

(d) Iteration 3

Figure 3. (a) Generator obtained from the Minkowski IFS, and first three iterations for $a = 1$, $b = 0.4$, and $\rho = 0.5$.



(a) ρ=0.2

(b) ρ=0.4

(c) ρ=0.6

(d) ρ=0.8

Figure 4. First iteration, M_1, for $b = a/3$ and four different values of ρ.

the results obtained from the FDTD simulation with those given by the commercial electromagnetic simulation suite, two interfaces were developed to have a single MFA which is independent of the simulation method. This not only provides flexibility (alternate geometries can be modeled by replacing the IFS in the algorithm) but also allows the user to easily change the size or shape of the structure by adjusting the input parameters.

Table 2. Coordinates of vertices for the geometry shown in Fig. 5.

Point	x	y	Point	x	y
1	−1.5	−1.5	11	1.5	1.5
2	−0.5	−1.5	12	0.5	1.5
3	−0.5	−1	13	0.5	1
4	0.5	−1	14	−0.5	1
5	0.5	−1.5	15	−0.5	1.5
6	1.5	−1.5	16	−1.5	1.5
7	1.5	−0.5	17	−1.5	0.5
8	1	−0.5	18	−1	0.5
9	1	0.5	19	−1	−0.5
10	1.5	0.5	20	−1.5	−0.5

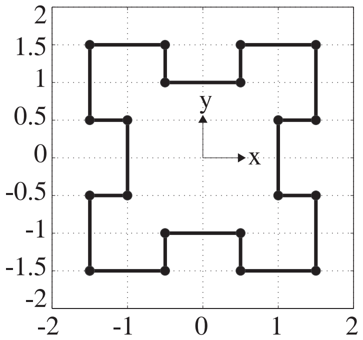


Figure 5. Output of Minkowski fractal algorithm (the dots correspond to the points listed in Table 2).

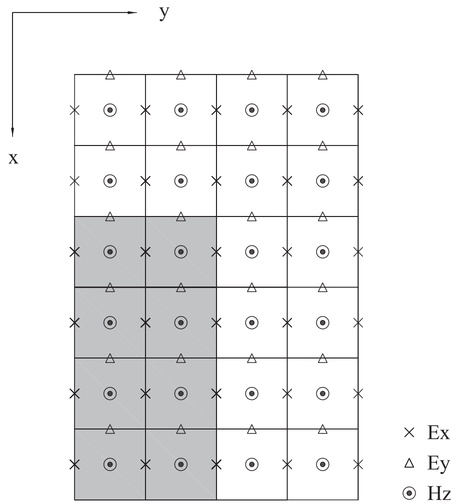


Figure 6. Location of E_x and E_y nodes at interfaces between the patch and the substrate.

3. INTERFACES TO ELECTROMAGNETIC SOLVERS

3.1. Interface to the Commercial EM Solver

In this case, the interface is written in the programming language specified by the commercial software. The nodes in the list are connected, in order, until a closed geometry is obtained. This process is illustrated for a first iteration M_1 with $a = 3$, $b = 1$, and $\rho = 0.5$. The coordinates of all 20 points needed to reproduce the geometry are given in Table 2, where the origin is located at the center of the structure; the vertices of the resulting patch are shown as dots in Fig. 5.

3.2. Interface to the in-house FDTD Code

The conducting patch is implemented as a perfect electric conductor (PEC) by setting the tangential components of the electric field that lie on the patch to zero. For those nodes that fall on an interface between the patch and the external medium (free space), one of the tangential components of the electric field, E_x or E_y, lies exactly on the interface and thus belongs to the PEC region, while the other one falls outside the patch geometry. Consequently, the PEC condition is only applied to the former. For the geometry shown in Fig. 6, the PEC condition is only applied to $E_x(E_y)$ for the free space cells located at the right (top) interface. An additional computer code was developed to include

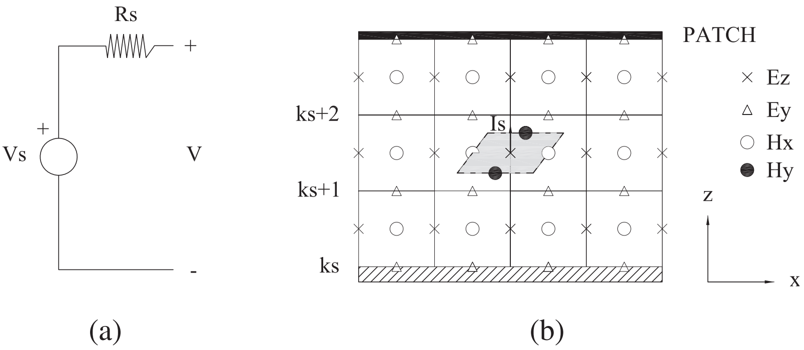


Figure 7. Equivalent circuit of the feed probe; (b) FDTD detailed geometry of gap-feed model used.

this algorithm. The code works as follows: first, all points contained in the circular list described earlier are mapped into a mesh defined by the resolution that will be used in the simulation, according to the specific requirements. The patch is no longer defined by its contour, and individual points are considered instead. The two tangential components of the electric field are examined separately and a decision algorithm extracts only those points that actually belong to the patch for each case. The main FDTD solver also includes a module that translates this information to the geometry of the problem according to the location of the patch. The implementation of the dielectric substrate with given electrical and physical parameters, the ground plane, the placement of the patch on the substrate, and the location of the source as a discrete port are also included in the interfaces to easily generate the entire geometry that is to be analyzed.

4. FDTD: DESIGN CONSIDERATIONS

The general FDTD method defines a rectangular lattice where each field component within a single cell is located at the midpoint of an edge (electric field components) or at the center of a cell face (magnetic field components). The constitutive material parameters are used to model the geometry of the problem by assigning the values for each cell throughout the geometry. In order to do so, all cells in the entire simulation domain are initially assigned a value corresponding to free space. Those cells that belong to a medium other than free space are then assigned the appropriate parameters. This is done individually for each field component, thus providing higher accuracy in the modeling for those cells where an edge is located at the interface between two different media. As mentioned earlier, the metallic sheets are modeled as PEC surfaces by assigning the tangential components of the electric field that lie on their surface a value of zero. A simple case would be a square PEC patch, located on the x-y plane, with dimensions $a \times a$. An easy way to accurately model the patch would be to assign dx and dy a value of $\delta = a/k$, where δ is an integer, and k is chosen so that the longest edge of an FDTD cell is short enough to provide good frequency resolution at the highest frequency on interest [10]. For the square patch, it is straightforward to match the grid to the dimensions of the patch, i.e., to make the boundaries fall exactly on cell edges, and only special care must be taken for those cells. The individual electric field components tangential to the patch, inlcuding those corresponding to edges that fall on its boundary, are set to zero, while the other components are updated according to the parameter that was previously assigned. Consequently, a square patch with a side length of a can be easily implemented in the FDTD code by 'counting' the number of field components that lie on the patch. The discrete port feed is implemented following the gap feed model given in [11], which is an extension of the one described in [12]. This model, depicted in Fig. 7, uses a voltage source V_s with an internal resistor, R_s, which is included to dissipate the energy that is reflected from the patch, thus reducing the number of steps needed for the FDTD run. The use of this resistor has been reported to reduce the necessary time steps by up to 32 times for this simple gap feed model [11].

5. SIMULATION RESULTS

The frequency response of the three geometries based on the Minkowski fractal have been analyzed by computing, for each of them, the return loss (S_{11}) at frequencies ranging from DC up to 20 GHz. A $55 \times 55 \times 0.795$ mm (length × width × thickness) dielectric slab with a relative permittivity of 2.2 is implemented as the substrate, on top of which the conducting patch is mounted. The patch, referred to as M_i where M stands for 'Minkowski' and i is the iteration number ($i = 0, 1, 2$), is modeled as a zero-thickness perfectly-conducting sheet as described in the previous section. The computational grid is defined by $\Delta_x = \Delta_y = 0.5$ mm, while the cell size along z is fixed to $\Delta_z = 0.265$ mm in order to have exactly three mesh cells inside the substrate along this direction. At the air-dielectric interface, the average value of the dielectric constant on either side is used for tangential fields. According to these parameters, the substrate takes $110 \times 110 \times 3$ cells, while the M_0 patch (located at the center of the substrate on the x-y plane) takes 80×80 cells. The ground plane is implemented as a zero-thickness, finite PEC sheet located directly underneath the substrate. The boundaries of the simulation domain are set 5 mesh cells away from the structure in all directions, and 12 perfectly matched layers (PML) based on the split-field approach discussed in [13–15] are used to absorb outgoing waves. There are 9 cells on top of the patch to simulate the vacuum region. With all these considerations, the simulation domain consists of a total of $122 \times 122 \times 28$ cells surrounded by the 12 perfectly absorbing layers. The results from the simulations for each geometry are given next.

5.1. Square Patch, M_0

A square ($40 \times 40\,\mathrm{mm}$) patch is obtained from the fractal algorithm for $i = 0$ by using a non-scaled version of the initiator to generate the closed geometry given by the four points corresponding to the vertices of the square. The FDTD interface produces a matrix with the information for all tangential electric field components within the extent of the patch. This matrix is used in the main code to apply the material parameters at the right nodes where the patch is located. The geometry of the imported patch is shown in Fig. 8, where the dot at the center of the structure represents the location of the feed. The cross-section along the dotted line shows the extent of the substrate and the location of the patch. The geometry displayed in the figure corresponds to the x the components of the electric field. The horizontal strips on either side of the patch correspond to the averaged values of the

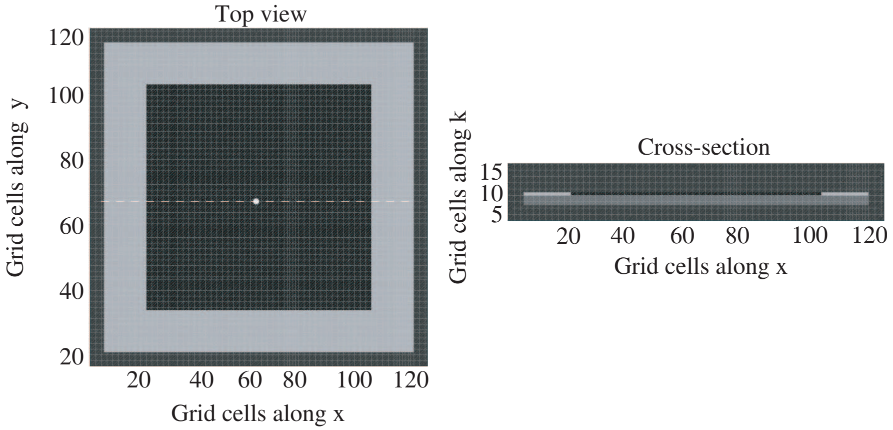
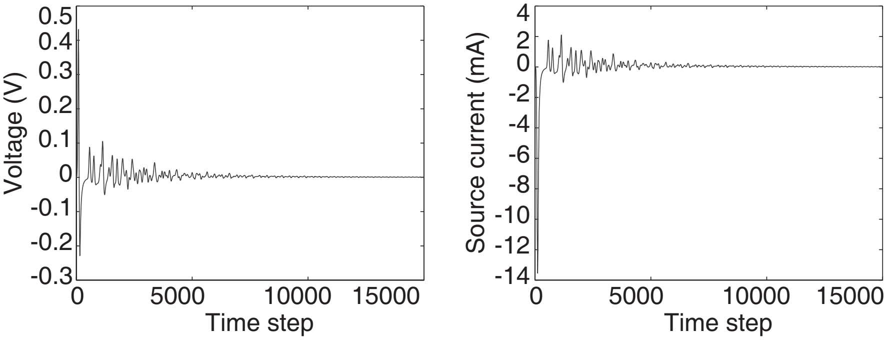


Figure 8. Geometry of the problem in the FDTD space: top view (left) and cross-section along dotted line (right).



(a) Input voltage

(b) Source current

Figure 9. Patch antenna source voltage (a) and current (b) for 15,000 time steps.

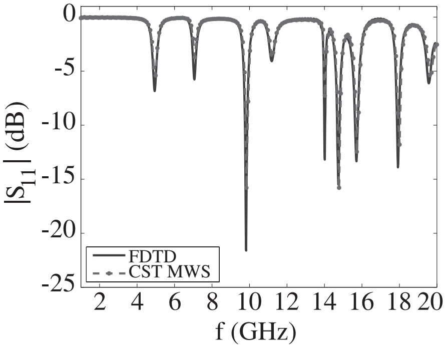


Figure 10. Frequency response (S_{11}) for M_0.

permittivity for E_x at the interface between the substrate and free space. The simulation was run for a total of 15,000 steps, and the steady-state source current and voltage are shown in Figs. 9(a) and (b), respectively. Fig. 10 shows the scattering parameter S_{11} given by the in-house FDTD code, represented by a solid line, which is compared to that obtained from CST MWS® (dotted line). Both plots are in very good agreement, and the locations of the operating resonances are highly correlated, with minor discrepencies in the peak values at resonant frequencies. This agreement is expected since the boundaries of the square patch, in both cases, can be aligned with the mesh. The importance of generating a mathematics-based geometry for fractals will become evident later when the first and second generation fractals are compared.

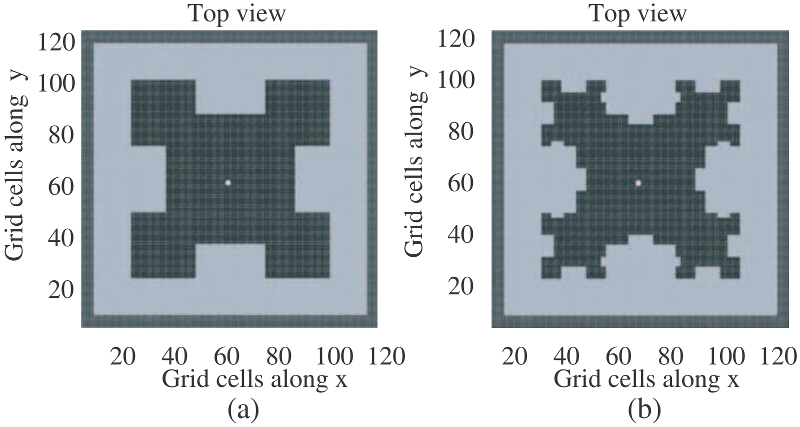


Figure 11. Geometry of the problem in the FDTD space for (a) M_1, and (b) M_2.

5.2. First (M_1) and Second (M_2) Iterations

As in the previous case, the geometry of the structure under study, already implemented in the FDTD solver, is shown in Fig. 11(a) and (b) for M_1 and M_2, respectively. The parameters of the initiator are $a = 40\,\text{mm}$, $b = a/3$, and $\rho = 0.5$. The source voltage and current (not shown) were also monitored to ensure steady-state, and the frequency response obtained for these fractal structures is shown in Fig. 12. The results still show good agreement between FDTD and CST MWS®, especially for M_1; as the edges of the structure attain a more complex shape for M_2, however, a greater discrepancy will result as shown in Fig. 12(b). This is due to a non-conformity of the structure to the generated mesh and the change in the electrical length of the fractal perimeter as compared to the actual structure. In such cases, conformal methods need to be implemented.

6. DISCUSSION

The effectiveness of the algorithms developed in this work (fractal generator and two interfaces) is evident from the good match between the results given by the two simulation codes. As stated in the previous section, some discrepancies at higher fractal iterations are due to the implementation of the perimeter of the patch, whose resolution eventually depends on the cell size. This can be explained by examining the particular implementation of the patch in the FDTD mesh: if the

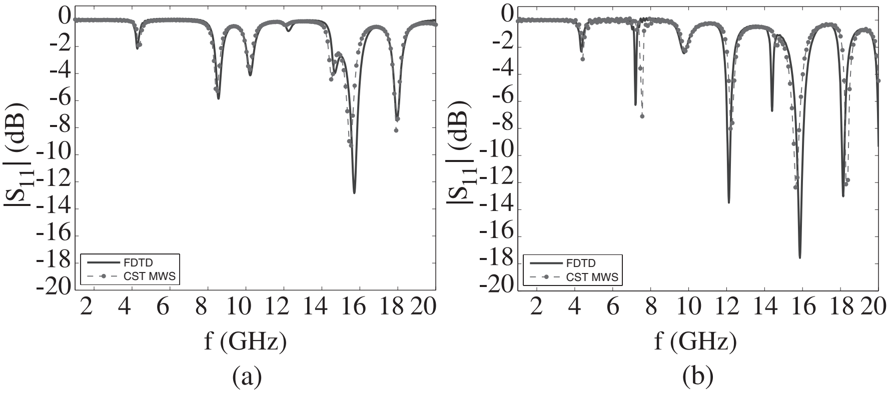


Figure 12. Compared frequency response for (a) M_1 and (b) M_2.

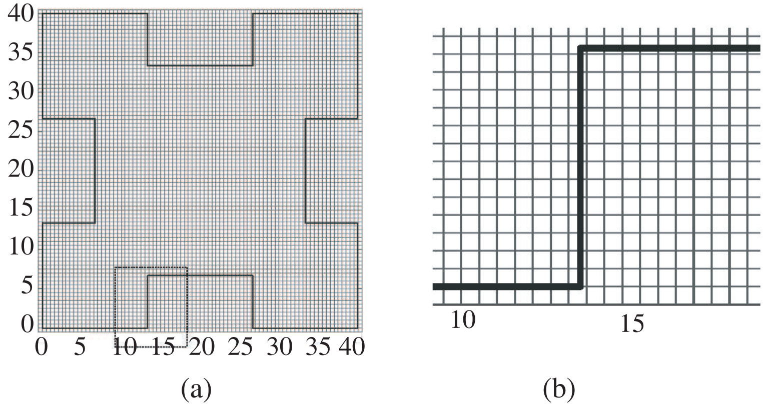


Figure 13. (a) Mesh representation of the geometry of the M_1 patch; (b) Detail of dotted region.

tangential electric field components (located exactly at the midpoint along the corresponding edge of a cell) lie on the patch, it is assumed that, as far as that component is concerned, the entire cell belongs to the patch, or to free space otherwise. The cells along the vertical and horizontal segments of an indentation as detailed in Fig. 13 show the loss of resolution for those cells that intersect a boundary (they will be interpreted by the code as completely filled with PEC or free space depending on the exact location of the midpoint of the edge with respect to the interface. Along the vertical segment, since both field components lie on the patch, the entire cell is implemented as

PEC for both E_x and E_y. For the horizontal segment, the E_x (E_y) component lies outside (inside) the patch, and the cell is modeled, for this component, as fully outside (inside). As a result, the gridded structure will be slightly larger or smaller (according to the cell size) than expected, and the perimeter of the patch will no longer match the actual one. This explains the frequency shifts observed in the results. This gridding error will have more implications in the results as the number of these 'special' cells increases. In these examples, there are 216 such cells for M_1, while this number rapidly increases to 432 with only one iteration. The higher the iteration number, the less likely that the fractal boundary will conform to the grid, thus introducing modeling errors. A proper gridding algorithm will minimize the error. Other ways to address these errors could include the use of a finer mesh at the cost of computer time and memory requirements, especially at higher frequencies. Sub-gridding or contour path finite-difference time-domain (CPFDTD) schemes have also been proposed to model PEC boundaries [16, 17]. Static mesh refinement techniques are well suited only when the structure involves Euclidean geometries and the mesh can be chosen to conform to the outer boundaries. For fractals, complex adaptive mesh refinement (AMR) techniques may be required [18]. A simpler approach which accurately models PEC surfaces without sub-gridding makes use of conformal finite-difference time-domain (CFDTD) algorithms with locally-conformal grids [19].

7. CONCLUSION

An electromagnetic solver based on the FDTD method, combined with a set of algorithms to implement complex geometries of radiating structures has been presented. Specifically, an algorithm to generate fractals and look into their radiation characteristics has been described. The fractals are derived from a Euclidean shape through a software-based etching process. The algorithm can also be used to generate fractals that can be applied to the commercial simulation software. The integration of IFS algorithms with two electromagnetic solvers, one of them being an in-house FDTD code with perfectly matched layers to truncate the simulation space, has been approached in this work. The algorithms have been tested for a microstrip patch antenna with several iterations of the Minkowski series. The results show that the software that was developed to analyze fractal structures can be used as a standalone tool to predict their radiation properties. The method highlighted in this paper could be a useful tool in simulation studies involving higher-order fractals.

REFERENCES

1. Yee, K., "Numerical solution of inital boundary value problems involving maxwell's equations in isotropic media," *IEEE Transactions on Antennas and Propagation*, Vol. 14, No. 3, 302–307, 1966.
2. Cohen, N., "Fractal antenna applications in wireless telecommunications," *Electronics Industries Forum of New England, Professional Program Proceedings*, 43–49, May 1997.
3. Mandelbrot, B. B., *The Fractal Geometry of Nature*, W. H. Freeman, Aug. 1982.
4. Werner, D. H. and R. Mittra, *Frontiers in Electromagnetics*, Wiley-IEEE Press, New York, 1999.
5. Vinoy, K., J. Abraham, and V. Varadan, "On the relationship between fractal dimension and the performance of multi-resonant dipole antennas using Koch curves," *IEEE Transactions on Antennas and Propagation*, Vol. 51, 2296–2303, Sep. 2003.
6. Gianvittorio, J. and Y. Rahmat-Samii, "Fractal antennas: A novel antenna miniaturization technique, and applications," *IEEE Antennas and Propagation Magazine*, Vol. 44, 20–36, Feb. 2002.
7. Barnsley, M., *Fractals Everywhere*, Academic Press, Boston, MA, 1988.
8. Peitgen, H. O., H. Jürgens, and D. Saupe, *Chaos and Fractals: New Frontiers of Science*, Springer, Feb. 1993.
9. Werner, D. and S. Ganguly, "An overview of fractal antenna engineering research," *IEEE Antennas and Propagation Magazine*, Vol. 45, 38–57, Feb. 2003.
10. Sarkar, T. K., S. M. Rao, and A. Djordjevic, "Electromagnetic scattering and radiation from finite microstrip structures," *IEEE Transactions on Microwave Theory and Techniques*, Vol. 38, 1568–1575, Nov. 1990.
11. Luebbers, R. and H. Langdon, "A simple feed model that reduces time steps needed for FDTD antenna and microstrip calculations," *IEEE Transactions on Antennas and Propagation*, Vol. 44, 1000–1005, Jul. 1996.
12. Luebbers, R., L. Chen, T. Uno, and S. Adachi, "FDTD calculation of radiation patterns, impedance, and gain for a monopole antenna on a conducting box," *IEEE Transactions on Antennas and Propagation*, Vol. 40, 1577–1583, Dec. 1992.
13. Berenger, J. P., "A perfectly matched layer for the absorption of electromagnetic waves," *Journal of Computational Physics*,

Vol. 114, No. 2, 185–200, 1994.

14. Berenger, J. P., "Three-dimensional perfectly matched layer for the absorption of electromagnetic waves," *Journal of Computational Physics*, Vol. 127, 363–379, 1996.
15. Berenger, J. P., "Making use of the PML absorbing boundary condition in coupling and scattering FDTD computer codes," *IEEE Transactions on Electromagnetic Compatibility*, Vol. 45, 189–197, May 2003.
16. Kunz, K. and L. Simpson, "A technique for increasing the resolution of finite-difference solutions of the maxwell equation," *IEEE Transactions on Electromagnetic Compatibility*, Vol. 23, 419–422, Nov. 1981.
17. Jurgens, T. and A. Taflove, "Three-dimensional contour FDTD modeling of scattering from single and multiple bodies," *IEEE Transactions on Antennas and Propagation*, Vol. 41, 1703–1708, Dec. 1993.
18. Dey, S. and R. Mittra, "A locally conformal finite-difference timedomain (FDTD) algorithm for modeling three-dimensional perfectly conducting objects," *IEEE Microwave and Guided Wave Letters*, Vol. 7, 273–275, Sep. 1997.
19. Yu, W. H. and R. Mittra, "A conformal FDTD algorithm for modeling perfectly conducting objects with curve-shaped surfaces and edges," *Microwave and Optical Technology Letters*, Vol. 27, No. 2, 136–138, 2000.